REMARKS/ARGUMENTS

The Applicant has carefully considered this application in connection with the Examiner's Action. The Applicant originally submitted Claims 1-20 in the application and makes no amendments to the claims in this response. Accordingly, Claims 1-20 are currently pending in the application. The Applicant respectfully requests reconsideration of this application in view of the foregoing amendments and the following remarks.

I. Formal Matters and Objections

The Examiner has objected to the drawings as being informal. The Applicant requests that the Examiner hold the objection in abeyance until such time as the Examiner indicates allowable subject matter.

The Examiner has objected to the specification as containing informalities, namely that the title of the invention is not descriptive. The Applicant has offered an amended title which is more clearly indicative of the invention to which the claims are directed.

II. Rejection of Claims 1-20 under 35 U.S.C. §103(a)

The Examiner rejected Claims 1-20 under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent 5,784,603 to Leung, *et al.*, in view of U.S. Patent 4,910,664 to Arizono, *et al.* This combination does not teach or suggest each and every element of the presently claimed invention, and is thus not a proper combination.

Claim 1 teaches, in relevant part, a loop recognizer that "reinstates the validity of said instructions in said loop, and prevents said prefetch circuitry from prefetching instructions outside of said loop until said loop completes execution." Paragraph 39 of the Specification shows that the loop recognizer "*disables* the prefetch circuitry 320 with respect to prefetching instructions outside the loop" (emphasis added). Furthermore, continuing in ¶39, "the loop recognizer 330 reinstates the validity of the instructions in the loop (via the cache line valid bit...)." When executing such a loop, a processor constructed according to the teachings of the present invention will not only confine any prefetch to instructions in the loop, but will advantageously disable a prefetch when the required instructions are resident in the instruction cache memory. This provides the advantage that prefetching of instructions is reduced during loop execution, reducing the consumption of power associated with the prefetch. In the extreme, when the loop is small enough to entirely fit within the instruction cache memory, no instruction prefetch at all is necessary after the first prefetch.

The Examiner cites Leung for the teaching of elements of Claim 1, including "a branch predictor that predicts whether a branch is to be taken," and "prefetch circuitry, coupled to said branch predictor, that prefetches instructions associated with said branch via said external memory interface if said branch is taken and prefetches sequential instructions via said external memory interface if said branch is not taken." By the Examiner's admission, Leung does not include the element of "a loop recognizer ... that determines whether a loop is present in fetched instructions and reinstates a validity of instructions in said loop and prevents said prefetch circuitry from prefetching instructions outside of said loop until said loop completes processing."

The Examiner then asserts that Arizono teaches the above-mentioned missing elements of Claim 1. However, a detailed reading reveals that Arizono does not teach this concept. Arizono

instead teaches a processor architecture designed to efficiently implement a "while" loop. Arizono is motivated by the desire to avoid prefetching instructions past the end of the loop that would otherwise be discarded when the program counter reaches the end of the loop and returns to the loop beginning address. Arizono specifically implements this teaching by storing the loop-end address in a register 8, and comparing the current program counter to the contents of the loop-end address register 8. When a match is detected, the prefetch counter 11 is reset to the loop-beginning address, previously stored in register 7. The prefetch of instructions then resumes at the loop-beginning address (col 5, lns 43-63). This has the effect of increasing the efficiency of the prefetch algorithm by preventing the prefetch of instructions past the end of the loop, but prefetch continues nevertheless. While improving prefetch efficiency somewhat, this implementation still requires the consumption of power to prefetch all instructions from slower, more power-consuming memory. Nowhere in Arizono is there a teaching or suggestion of the element of "a loop recognizer ... that determines whether a loop is present in fetched instructions and reinstates a validity of instructions in said loop and prevents said prefetch circuitry from prefetching instructions outside of said loop until said loop completes processing."

For the above-mentioned reasons, Leung and Arizono fail to teach or suggest all the limitations of Claim 1. Furthermore, neither reference contains a suggestion that the validity of prefetched instructions can or should ever be reinstated. Therefore the rejection of Claim 1 over the combination of Leung and Arizono is improper. By similar argument, the rejection of independent Claims 8 and 15, and the claims that depend from Claims 1, 8 and 15 is improper. The Applicant therefore respectfully requests that the Examiner withdraw the rejections on all claims.

III. Conclusion

In view of the foregoing amendments and remarks, the Applicant now sees all of the claims currently pending in this application to be in condition for allowance and therefore earnestly solicits a Notice of Allowance for Claims 1-20.

The Applicant requests the Examiner to telephone the undersigned attorney of record at (972) 480-8800 if such would further or expedite the prosecution of the present application.

Respectfully submitted,

HITT GAINES, P.C.

David H. Hitt
Registration No. 33,182

SEPTEMBER 22, 2004
Dated: September 22, 2004

P.O. Box 832570
Richardson, Texas 75083
(972) 480-8800